Survey on various classification techniques in data mining

[1]Sonia Jain, [2]Avinash Dhole

[1]Student , [2]Assistant Professor,
[1]Computer Science & Engineering,
[1]Raipur Institute of Engineering, Raipur, India

Abstract— Classification is a known as a supervised learning process in which we have a prior information about the data to be classified. It is used to find out the relation between the dataset and the group which is to be checked with the dataset by comparing its each instance. The most famous classification algorithms which are used nowadays are k-nearest neighbor, C4.5, ID3, SVM and Random forest. Statistical approach, Neural networks and MAchine learning approaches are followed by the general classification algorithms. And taking into consideration of the above mentioned approaches the following paper introduces us to the survey of the aforementioned algorithms in details including their advantages and disadvantages.

Index Terms— **supervised learning, K-nearest neighbor, C4.5, ID3, SVM, Random Forest.**

I. INTRODUCTION

As a kind of inductive learning algorithm, decision tree algorithms have been successful to build classifiers with the aim to maximize the classification accuracy. The well-known ID3 [1], C4.5 [2], CART [3], and so on all center around inducing decision trees for the high classification accuracy. However, one of the main difficulties of tree building in practice is that the majority of variables tests have associated cost, which may be diverse for each test [5,6]. Since data is not free, instead of only focusing on classification accuracy, a learner should perform an economic yet effective induction in practical application. Test-cost sensitive learning is more practical than simple traditional classification in many applications such as intelligent medical diagnostic systems [7]. To the simplest of our information, some existing test-cost sensitive learning algorithms ar regarding equalization the act of two styles of price, particularly the misclassification price and also the take a look at price, to determine that take a look at are done [8–13]. When a test case is classified by a decision tree, some algorithms have tried to find a tradeoff between the accuracy and the test cost. These algorithms are all the improved test-cost sensitive versions based on ID3 or C4.5 and they directly adapt existing information theoretic measures by including costs. All these algorithms reduce the test cost, unfortunately, yet at the same time degrade the classification accuracy.

In this paper we focus on building decision trees which have not only the lower test cost but also the higher classification accuracy. We can build decision trees which reach the same classification accuracy as C4.5, mean while reduce the test cost significantly. Previous works reduce the test cost while also degrade the classification accuracy. When selecting the current attribute to build a tree, we cannot only consider the total test cost, but also the classification accuracy.

The rest of this paper is organized as follows. Section 2 introduces some related works on attribute selection measures in decision tree learning and test-cost sensitive decision tree. Section 3 proposes the conclusion and comparison among the following algorithms ID3 , K nearest algorithm, C4.5 algorithm, SVM and Random Forest algorithm.

II. RELATED WORK

In this section previous related work will be explain attribute selection measures in decision tree learning and Test-cost sensitive decision tree.

Neural Network

The field of Neural Networks has arisen from various sources starting from understanding and emulating the human brain to broader problems with repeating human skills like speech and might be use in varied fields like banking, legal, medical, news, in classification program to reason information as intrusive or traditional. typically neural networks comprises layers of interconnected nodes wherever every node manufacturing a non-linear operate of its input and input to a node could return from different nodes or directly from the input file. Also, some nodes area is known with the output of the network.

On the premise of this instance there area unit completely different applications for neural networks that involve recognizing patterns and creating straightforward choices concerning them. In airplanes we will use a neural network as a basic autopilot wherever input units reads signals from the assorted cockpit instruments and output units modifying the plane's controls befittingly to stay it safely on the right track. within a manufacturing plant we will use a neural network for internal control.

Classification Algorithm

Classification is one amongst the information Mining techniques that's in the main accustomed analyze a given dataset and takes every instance of it and assigns this instance to a selected category specified classification error are going to be least. it's accustomed extract models that accurately outline vital information categories at intervals the given dataset. Classification could be a 2 step method. throughout start the model is formed by applying classification formula on coaching information set then in second step the extracted model is take a look ated against a predefined test dataset to live the model trained performance and accuracy. Thus classification is that the method to assign category label from dataset whose class label is unknown.

ID3 Algorithm

Id3 calculation starts with the first set because the root hub. On each cycle of the rule it emphasizes through each unused attribute of the set and figures the entropy (or information obtain IG(A)) of that attribute. At that time chooses the attribute that has the littlest entropy (or biggest information gain) worth. The set is S then split by the chosen attribute to provide subsets of the knowledge. The rule yield to recourse on every and each item in set and considering solely things ne'er chosen before. Formula on a set might waken a halt in one amongst these cases:

- Every element in the subset belongs to the same category (+ or -), then the node is become a leaf and tagged with the category of the examples.
- If there are no more attributes to be selected but the examples still don't belong to an equivalent category then the node is changed into a leaf and tagged with the foremost common category of the examples therein set.
- If there are no examples in the subset, then this happens when parent set found to be matching a specific value of the selected attribute. For example if there was no example matching with marks >=100 then a leaf is created and is labelled with the most common class of the examples in the parent set.

Working steps of algorithm is as follows,

- Use the data set S and calculate the entropy for each attribute.
- Using the attribute for which entropy is minimum and split the set S
- Construct a decision tree node containing that attribute in a dataset.
- Recurse on each member of subset using remaining attributes.

SVM

SVM have attracted a great deal of attention in the last decade and actively applied to various domains applications. SVMs are typically used for learning classification, regression or ranking function. SVM are based on statistical learning theory and structural risk minimization principle and have the aim of determining the location of decision boundaries also known as hyperplane that produce the optimal separation of classes. Maximizing the margin and thereby creating the largest possible distance between the separating hyperplane and the instances on either side of it has been proven to reduce an upper bound on the expected generalization error. Efficiency of SVM based classification is not directly depend on the dimension of classified entities. Though SVM is the most robust and accurate classification technique, there are several problems. The data analysis in SVM is based on convex quadratic programming, and it is computationally expensive, as solving quadratic programming methods require large matrix operations as well as time consuming numerical computations. Training time for SVM scales quadratically in the number of examples, so researchers strive all the time for more efficient training algorithm, resulting in several variant based algorithm.

SVM can also be extended to learn nonlinear decision functions by first projecting the input data onto a high-dimensional feature space using kernel functions and formulating a linear classification problem in that feature space [4]. The resulting feature space is much larger than the size of dataset which are not possible to store in popular computers. Investigation on this issues leads to several decomposition based algorithms. The basic idea of decomposition method is to split the variables into two parts: set of free variables called as working set, which can be updated in each iteration and set of fixed variables, which are fixed at a particular value temporarily. This procedure is repeated until the termination conditions are met[5]. Originally, the SVM was developed for binary classification, and it is not simple to extend it for multi-class classification problem. The basic idea to apply multi classification to SVM is to decompose the multi class problems into several two class problems that can be addressed directly using several SVMs.

K Nearest Algorithm

The nearest neighbor rule distinguishes the categorification of unknown datum on the idea of its nearest neighbor whose class is already known M. cover and P.E.The purpose k nearest neighbour (KNN) during which nearest neighbor is computed on the idea of estimation of k that indicates what percentage nearest neighbors ar to be thought-about to characterize category of a sample datum.. It makes utilization of the quite one nearest neighbor to see the category during which the given information belongs to and and consequently it's referred to as as KNN. These knowledge samples are required to be within the memory at the run time and therefore they're observed as memory-based technique.The training points area unit allotted weights in step with their distances from sample datum. however at an equivalent time the machine quality and memory necessities stay the first concern faithfully.To overcome memory limitation size of data set is reduced. For this the repeated patterns which don't include additional data are also eliminated from training data set.

To more enhance the focuses that don't influence the result ar in addition eliminated from coaching data set. The NN coaching information set may be organized utilizing totally different systems to reinforce over memory limit of KNN. The KNN implementation may be done victimisation ball tree, k-d tree, nearest feature line (NFL), axis search tree and orthogonal search

tree. The tree structured coaching information is more divided into nodes and techniques like NFL and tunable metric divide the coaching information set in line with planes. victimisation these algorithmic programs we will expand the speed of basic KNN algorithm. take into account that AN object is sampled with a collection of various attributes. assumptive its cluster may be determined from its attributes; totally different algorithms may be wont to automatise the classification method. In pseudo code k-nearest neighbor classification algorithmic program may be expressed.

K →number of nearest neighbors For each object Xin the test set do calculate the distance D(X,Y) between X and every object Y in the training set
neighborhood ! the k neighbors in the training set closest to X
X.class → Select Class (neighborhood) End for

C 4.5 Algorithm

C4.5 is an algorithm used to produce a decision tree which is an expansion of prior ID3 calculation. It enhances the ID3 algorithm by managing both continuous and discrete properties, missing values and pruning trees after construction. The decision trees created by C4.5 can be used for grouping and often referred to as a statistical classifier. C4.5 creates decision trees from a set of training data same way as Id3 algorithm. As it is a supervised learning algorithm it requires a set of training examples which can be seen as a pair: input object and a desired output value (class). The algorithm analyzes the training set and builds a 15

Classifier that must have the capacity to accurately arrange both training and test cases. A test example is an input object and the algorithm must predict an output value. Consider the sample training data set $S=S_1,S_2,...Sn$ which is already classified. Each sample S_i consist of feature vector $(x_{1,i}, x_{2,i}, ..., x_{n,i})$ where x_j represent attributes or features of the sample and the class in which S_i falls. At each node of the tree C4.5 selects one attribute of the data that most efficiently splits its set of samples into subsets such that it results in one class or the other. The splitting condition is the normalized information gain (difference in entropy) which is a non-symmetric measure of the difference between two probability distributions *P* and *Q*. The attribute with the highest information gain is chosen to make the decision. General working steps of algorithm is as follows,

- Assume all the samples in the list belong to the same class. If it is true, it simply creates a leaf node for the decision tree so that particular class will be selected.
- None of the features provide any information gain. If it is true, C4.5 creates a decision node higher up the tree using the expected value of the class.
- Instance of previously-unseen class encountered. Then, C4.5 creates a decision node higher up the tree using the expected value.

Random Forest Algorithm

Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error for forests converges a.s. to a limit as the number of trees in the forest becomes large

The common element in all of these procedures is that for the kth tree, a random vector Θk is generated, independent of the past random vectors Θ1, ... ,Θk−1 but with the same distribution; and a tree is grown using the training set and Θk , resulting in a classifier h(x,Θk) where x is an input vector. For instance, in bagging the random vector Θ is generated as the counts in N boxes resulting from N darts thrown at random at the boxes, where N is number of examples in the training set. In random split selection Θ consists of a number of independent random integers between 1 and K. The nature and dimensionality of Θ depends on its use in tree construction. After a large number of trees is generated, they vote for the most popular class. We call these procedures random forests.

The random forests algorithm applies the general technique of bagging, to tree learners. Given a training set $X = x_1, ..., x_n$ with responses $Y = y_1, ..., y_n$, bagging repeatedly (*B* times) selects a random sample with replacement of the training set and fits trees to these samples:

For $b = 1, ..., B$:

1. Sample, with replacement, *B* training examples from *X*, *Y*; call these X_b, Y_b.
2. Train a decision or regression tree f_b on X_b, Y_b.

After training, predictions for unseen samples x' can be made by averaging the predictions from all the individual regression trees on x':

III. CONCLUSION

This paper focuses on various classification techniques A new test-cost sensitive decision tree learning algorithm is proposed in this paper, which aims to keep the high classification accuracy meanwhile reduce the total test cost. Compared to C4.5, ID3, K nearest algorithm and ,existing test-cost sensitive decision tree learning algorithms by adapting information theoretic measures to introduce the test cost degrade the classification accuracy when they reduce the total test cost, while our algorithm maintains the same classification accuracy as C4.5 at the same time significantly reduces the total test cost. This paper provides a new idea for research, i.e., it does not has to reduce the test costs at the cost of the loss of classification accuracy. We can reduce the total test cost and maintain the same classification accuracy as C4.5 simultaneously. For this purpose, A random factor is introduced to tree building to make trees more so that a random attribute selection measure is presented. Instead of the greedy strategy, the proposed random attribute selection measure employs a random strategy to guide the selection of the optimal attribute for splitting. Our current version transforms the test-cost sensitive classification problem into a constrained single-objective optimization problem. However, in many real-world applications, continuous features are widespread and, therefore, extending it to directly handle applications with continuous features is another direction for our future study.biased in favor of the test cost or the classification accuracy

IV. ACKNOWLEDGMENT

I am highly indebted to Mr Avinash Dhole Sir for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.
I would like to express my gratitude towards my parents & member of Raipur Institute Technology (CSVTU) for their kind cooperation and encouragement which help me in completion of this project.

References

[1.] J.R. Quinlan, J. Ross, Induction of decision trees, Mach. Learn. 1 (1) (1986)81–106.
[2.] J.R. Quinlan, C4.5: Programs for Machine Learning, San Mateo, MorganKaufmann, 1993.
[3.] L. Breiman, J. Friedman, J. Stone, Classification and Regression Trees, CRCPress, 1984.
[4.] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[5.] C. Elkan, The foundations of cost-sensitive learning, in: International JointConference on Artificial Intelligence, Lawrence Erlbaum Associates, 2001, pp.973–978.
[6.] J.R. Quinlan, P.J. Ross, Inductive knowledge acquisition: a case study, in:Proceedings of the Second Australian Conference on Applications of ExpertSystems, Addison-Wesley Longman Publishing Co., 1987, pp. 137–156.
[7.] V. López, D. Rianõ, J.A. Bohada, Improving medical decision trees by combiningrelevant health-care criteria, Expert Syst. Appl. 39 (14) (2012) 11782–11791.
[8.] P.D. Turney, D. Peter, Cost-sensitive classification: empirical evaluation of ahybrid genetic decision tree induction algorithm, J. Artif. Intell. Res. (1995)369–409.
[9.] C.X. Ling, X. Charles, Decision trees with minimal costs, in: Proceedings of the21st International Conference on Machine learning, ACM, 2004.
[10.] Z. Qin, S. Zhang, C. Zhang, Cost-sensitive decision trees with multiple costscales., in: AI 2004: Advances in Artificial Intelligence, Springer BerlinHeidelberg, 2005, pp. 380–390.
[11.] T. Wang, Handling over-fitting in test cost-sensitive decision tree learning byfeature selection, smoothing and pruning, J. Syst. Softw. 83 (7) (2010)1137–1147.
[12.] F. Min, W. Zhu, A competition strategy to cost-sensitive decision trees., in:Rough Sets and Knowledge Technology, Springer Berlin Heidelberg, 2012, pp.359–368.
[13.] Y. Weiss, Y. Elovici, L. Rokach, The cash algorithm-cost-sensitive attributeselection using histograms, Inf. Sci. 222 (2013) 247–268.
[14.] F. Min, H. He, Y. Qian, W. Zhu, Test-cost-sensitive attribute reduction, Inf. Sci.181 (22) (2011) 4928–4942.
[15.] F. Min, Q. Hu, W. Zhu, Feature selection with test cost constraint, Int. J.Approx. Reason. 55 (1) (2014) 167–179.